Provide Sample Python Program for all the below 10 exercises.

Instructions:

- Your answers should be provided in one single PDF document.
- It should contain the program you wrote(python) and should be clear enough to go through the code
  (proper code comments and explanations on each code you wrote should be there)
- The output you received for each of the exercise should be attached as well in your answer sheet

Exercise 1: Library Management System

Create a Book class with attributes such as title, author, and isbn. Implement a Library class that manages a collection of books. Include methods to add, remove, and search for books by title or author. Store the book details in a text file.

Exercise 2: Student Grades System

Create a Student class with attributes like name, id, and a list of grades. Implement methods to add grades, calculate the average, and determine if the student passed (average >= 60). Store student details in a JSON file.

Exercise 3: Task Manager

Implement a Task class with attributes title, description, and status. Create a TaskManager class that uses a queue to manage tasks (add, complete, and display pending tasks). Save the task list to a CSV file.

Exercise 4: Shopping Cart

Create a Product class with attributes name, price, and quantity. Implement a ShoppingCart class that allows adding/removing products and calculating the total price. Store the cart contents in a dictionary.

Exercise 5: Movie Database

Create a Movie class with attributes title, genre, and rating. Implement a MovieDatabase class that can add movies, search by genre, and display all movies sorted by rating. Use a set to store unique genres.

Exercise 6: Phonebook

Create a Contact class with attributes name, phone, and email. Implement a Phonebook class that allows adding, removing, and searching for contacts. Store contacts in a dictionary, and save/load the phonebook from a text file.

Exercise 7: Stack Implementation

Implement a Stack class with methods to push, pop, and check if the stack is empty. Write a function that checks if a given string of parentheses is balanced using your Stack class.

Exercise 8: Queue Implementation

Implement a Queue class with methods to enqueue, dequeue, and check if the queue is empty. Create a function that simulates a ticketing system where customers join the queue and are served in order.

Exercise 9: Recipe Book

Create a Recipe class with attributes name, ingredients, and instructions. Implement a RecipeBook class that allows adding, removing, and searching for recipes. Store recipes in a JSON file.

Exercise 10: Weather Data Analyzer

Create a WeatherData class to store temperature data for a week. Implement methods to calculate the average temperature, highest, and lowest temperatures. Store data in a tuple and read/write from/to a CSV file.